A Wideband Digital Pseudo-Gaussian Noise Generator

W. J. Hurd
Communications Systems Research Section

A digital system has been constructed for the generation of wideband gaussian noise with a spectrum which is flat to within ±0.5 dB from 0 to 10 MHz. These characteristics are substantially better than those of commercially available analog noise generators, and are required in testing and simulation of wideband communications systems. The noise is generated by the analog summation of thirty essentially independent binary waveforms, clocked at 35 MHz, and low-pass filtered to 10 MHz.

I. Introduction

A digital system has been constructed for the generation of wideband gaussian noise with a spectrum which is flat to within ±0.5 dB from 0 to 10 MHz. This is substantially better than other noise sources now available. Noise of this bandwidth and spectral flatness is required in the testing and simulation of communications systems whose performance in the presence of noise must be accurately known. For example, by doing this accurate testing, margins of uncertainty are reduced, so that data rates can increase. The first application of this noise generator was in the evaluation of a prototype model of a code-regenerative spacecraft ranging transponder, the ranging clean-up loop for outer planet use.

II. General Description

Wideband video gaussian noise is required to test communications systems and in the simulation of many other types of systems. The required noise bandwidths may range from zero or close to zero Hertz to several Megahertz. Commercially available analog noise generators, using noise tube or diode sources, are often not satisfactory for several possible reasons: the bandwidth may not be wide enough; the spectral density may not be flat enough in the passband; the probability distribution may not be close enough to gaussian; and the stability of the noise parameters may not be good enough. The amplitude distribution is often so skewed that the sample waveform is visibly asymmetric when viewed on an oscilloscope, and the spectral density is typically specified as flat to within only ±1 or ±3 dB. Noise with these characteristics is clearly not acceptable in testing a communication system whose performance must be known to within one or two tenths of a decibel.

Recently, noise generators have become available which use digital logic to generate pseudo-random binary waveforms. The binary waveforms are then low-pass filtered to obtain quasi-gaussian noise. The principle is that according to the Central Limit Theorem, the distribution of the sum of a number of independent random variables tends toward gaussian as the number of terms in the sum is increased. The low-pass filtering performs the required summing. The problem with this type of system is that the

distribution of the pseudonoise becomes more gaussian only as the filter bandwidth is made progressively smaller with respect to the clock rate of the binary sequence. For some applications, the noise may be close enough to gaussian only if the clock rate is hundreds of times higher than the filter bandwidth, which means that the noise bandwidth will be much less than the digital clock rate—perhaps 100 kHz if the clock rate is 35 to 50 MHz.

Since the closeness of the distribution to gaussian depends on the number of independent terms in a summation, we observe that these terms can be generated either sequentially, as above, or in parallel. For example, the outputs of a large number of different binary pseudo-noise generators could be summed, and then low-pass filtered only enough to reduce the transient effects of the clocking and to shape the spectrum. It is this general principle which is used in the noise generator described here. One major improvement is made, however, which considerably reduces the hardware: a large number of different shifts of the same sequence are generated simultaneously, with each additional shift requiring only one simple integrated circuit, not an entire new shift register. The different shifts are essentially uncorrelated, and are far enough apart in time so that no shift contributes more than once to any experiment. The actual system constructed is clocked at 35 MHz, and the quasi-gaussian noise has a spectrum which is flat to ± 0.5 dB from 0 to 10 MHz.

III. Algebraic Theory

Suppose we have two maximal length linear shift register ($pn-$) sequences generated in shift registers X and Y of relatively prime lengths N and M, respectively. The sequences at the various shift register stages are labeled $X_i(k)$ and $Y_j(k)$ where the subscripts $i = 0, 1, \cdots, N-1$ and $j = 0, 1, \cdots, M-1$ denote the register stages, the argument k denotes time, and the binary values are taken to be $+1$ and -1. The shifting is from higher to lower numbered stages, so

$$X_i(k+1) = X_{i+1}(k), \qquad k = 0, 1, \cdots, N-2$$
$$Y_j(k+1) = Y_{j+1}(k), \qquad k = 0, 1, \cdots, M-2$$

and $X_{N-1}(k+1)$ and $Y_{M-1}(k+1)$ are linear functions of the values in the respective registers at time k.

We can now form the sequences $Z_{ij}(k) = X_i(k)\,Y_j(k)$. The periods of the X, Y, and Z sequences are $p_x = 2^N - 1$, $p_y = 2^M - 1$, and $p_z = p_x p_y = 2^{N+M} - 2^N - 2^M + 1$. Note that for reasonably large N and M, p_z is almost equal to the maximum length of a linear sequence generated by a shift register of $N + M$ stages.

The correlation properties of Z are similar to those of X and Y, so that Z is approximately a white noise sequence. The normalized in phase correlations of X and Y are 1, and the out of phase correlations are $-1/p_x$ and $-1/p_y$, respectively. For Z, the in phase correlation is 1, and the out-of-phase correlation is $+1/p_z$, except for phase shifts np_x and $np_y \pmod{p_z}$, at which points it is $-1/p_y$ and $-1/p_x$, respectively. For reasonably large p_x and p_y, all of the out of phase correlations are small, as desired for "white" noise. The cycle and add property of pn-sequences also extends to the product sequences, with similar exceptions as for the correlation properties.

One property of pn-sequences which does not extend to the product sequence is that of having all of the p_x (or p_y) possible non-trivial N (or M)-tuples occur exactly once in the cycle of length p_x (or p_y). Almost all possible $(N + M)$-tuples occur in Z, however. There are $2^{N+M} - 1$ possible nontrivial $(N + M)$-tuples in a maximum length linear sequence from a shift register of N stages, but the period of Z is $2^N + 2^M - 2$ less than this. Hence all but $2^N + 2^M - 2$ of the nontrivial $(N + M)$-tuples occur. The number not occurring is relatively small if N and M are moderately large, so that the effect on the randomness properties of the sequence is negligible.

The phase relationships between the $Z_{ij}(k)$ can be determined by the phase of each with respect to a reference sequence, which we choose to be $Z_{oo}(k)$. Denoting the delay from Z_{ij} to Z_{oo} by t_{ij}, we have $Z_{ij}(k) = Z_{oo}(k + t_{ij})$ for all k. This requires that $X_i(k) = X_o(k + t_{ij})$ and $Y_j(k) = Y_o(k + t_{ij})$, both for all k. Since

$$X_i(k) = X_0(k + i + np_x)$$

for all n and k, and

$$Y_j(k) = Y_o(k + j + mp_y)$$

for all m and k, we have

$$t_{ij} \equiv i \pmod{p_x}$$
$$\equiv j \pmod{p_y}$$

Now, since p_x and p_y are relatively prime, we can use the Euclidean algorithm to find a_x and a_y such that

$$a_x p_x + a_y p_y = 1$$

Finally,

$$t_{ij} \equiv i\,a_x p_x + j a_y p_y \,(\mathrm{mod}\, p_x p_y)$$

For the noise generator constructed, N and M were chosen to be 41 and 23, respectively. For this case,

$$a_x = 2^{21} + 2^{18} + 2^{16} + 2^{13} + 2^{11} + 2^8 + 2^6 + 2^3 + 2^0$$

and

$$a_x p_x = 1 - a_y p_y = a_x (2^{41} - 1)$$

Expressed as binary fractions,

$$a_x p_x / p_z \approx 0.0100101001 \cdots$$

which is approximately a repeating fraction equal to 9/31. Thus

$$t_{ij} \approx \frac{9}{31}(i - j)\, p_z$$

with the approximation being valid for small enough i and j that the repeating fraction approximation is good. We see from this that by carefully selecting pairs i and j, we can obtain 31 shifts of Z_{ij} which are approximately equally spaced modulo p_z.

In the actual implementation, 30 such shifts were used. These shifts could have been chosen by fixing j and choosing 30 consecutive taps of X. This has the disadvantage, however, that although the Z_{ij} thus obtained would be uncorrelated, adjacent sums of the thirty Z_{ij} are highly dependent, since 29 of the 30 terms in adjacent sums would be the same, except possibly for the sign. This is seen by expanding the sums and comparing, for example,

$$\sum_{i=0}^{29} X_o(k)\, Y_i(k) = X_o(k)\,[Y_o(k) + \sum_{i=1}^{29} Y_i(k)]$$

and

$$\sum_{i=0}^{29} X_o(k+1)\, Y_i(k+1) = X_o(k+1) \sum_{i=0}^{29} Y_{i+1}(k)$$

$$= X_o(k+1)\,[Y_{31}(k) + \sum_{i=1}^{29} Y_i(k)]$$

These two sums differ only by $Y_o(k) - Y_{31}(k)$ when $X_o(k) = X_o(k+1)$, and are similarly related but of opposite sign when $X_o(k) = -X_o(k+1)$. For this reason, care was taken to use different X_i for each Z, and to use no Y_j more than twice. Some duplication of Y_j's was necessary, because only 21 were available. The actual Z_{ij} selected correspond to $(i, j) = (n, 21 - n)$, $n = 0, 1, \cdots, 21$, and $(i, j) = (30, 7)$, $(31, 6)$, $(28, 1)$, $(23, 4)$, $(36, 5)$, $(35,2)$, $(38, 3)$, and $(37, 0)$.

With a system clock rate of 35 MHz, Z repeats approximately every 17000 years, and each pair of Z_{ij}'s is separated by over 500 years.

The shift register sequences are defined by the polynomials

$$x^{41} + x^3 + x^2 + x + 1 = 0$$

and

$$y^{23} + y^{17} + y^{11} + y^5 + 1 = 0$$

The polynomial for X was selected because the nonzero terms are consecutive, which means that implementation can be accomplished by toggling three of the stages according to the inputs to these stages, instead of implementing the multiplication. The polynomial for Y was chosen for a more important reason. This is that this polynomial divides no trinomials of degree less than 500. This algebraic property tends to lessen any possible skewedness of the distribution of sums of consecutive terms in the sequence (Ref. 1). To eliminate any residual skewedness, half of the thirty Z_{ij}'s generated are inverted in the final summation. Since their distributions are identical, this symmetrizes the distribution of the sum.

IV. Implementation

A block diagram of the noise generator is shown in Fig. 1. The digital portion is implemented entirely out of dual-in-line integrated circuit flip-flops, primarily because higher speed circuitry is available in flip-flops than in shift registers, and because access to both outputs of most stages was required. Since logical "1" and "0" in the digital circuitry correspond to -1 and $+1$, respectively, of the previous section, the EXCLUSIVE-OR operation corresponds to multiplication. The multiplications of the X_i and Y_j are thus implemented using AND-OR input flip-flops connected to perform EXCLUSIVE-OR's. This technique is also used to generate the feedback function for the Y register. For the X register, it was only necessary

to connect three stages so as to toggle when the previous stage is logically "1," instead of shifting, because ot the previously mentioned property of the polynomial.

The outputs of the Z_{ij} are summed using a resistor network at the input to a wideband operational amplifier. At the amplifier output, the signal is filtered by a parallel tank circuit to eliminate the clock frequency, and passed through a 50-Ω resistor to provide a standard 50-Ω source impedance. When loaded with 50 Ω, the spectrum of the system output is flat to within about ±0.5 dB from zero to 10 MHz.

It is planned to increase the clock rate of the system to about 50 MHz by replacing the current integrated circuits with faster but PIN-compatible circuits. When this is accomplished, the output spectrum will be further shaped by filtering to obtain the flattest spectrum possible, and statistical tests of the noise properties will be undertaken.

Reference

1. Lindholm, J. H., "An Analysis of the Pseudo-Randomness Properties of Subsequences of Long m-Sequences," *IEEE Trans. Info. Theory*, Vol. IT-14, pp. 569–576, July 1968.

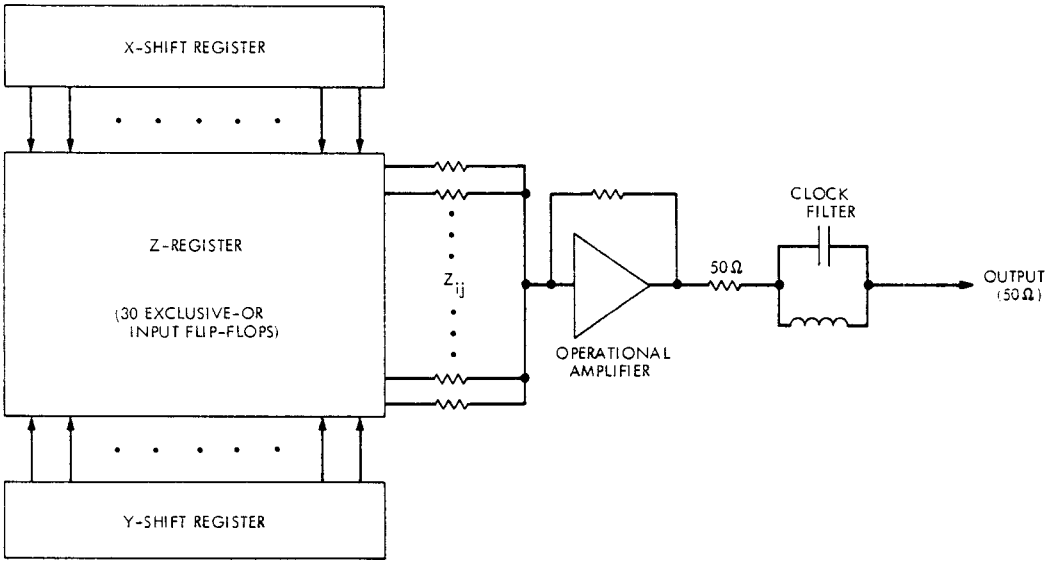

Fig. 1. Noise generator block diagram